OntoURIResolver: URI Resolution Service using Multiple Ontologies

Pyung Kim[1], Taehong Kim[1,2], Seungwoo Lee[1], Hanmin Jung[1], Won-Kyung Sung[1]

[1] Knowledge Information Center,
Korea Institute of Science and Technology Information, Korea
[2] Practical Information Science,
University of Science & Technology, Korea
{pyung, kimtaehong, swlee, jhm, wksung}@kisti.re.kr

Abstract. To make a local ontology interoperable and sharable with LOD, we have to make linkages URIs between local ontology and LOD. In order to decide appropriate URI for specific entity, OntoURIResolver collects RDF (Resource Description Framework) triples of multiple ontologies by SPARQL, divides URIs into several groups by comparing RDF triples of URIs and recommends a canonical URI and entity name for each group using statistics of RDF triples. We experiment comparison of sameas.org and OntoURIResolver with top 10 ranked authors of DBLP. Users can find a specific URI for entity and make interconnections with LOD to maximize the effectiveness of ontology through this service.

Keywords: OntoURIResolver, URI Resolution, Ontology, Linked Open Data

1 Introduction

Ontology is a formal representation of the knowledge by a set of concepts within a domain and the relationships between those concepts. In order to share, reuse and unify data set, we use ontology as a knowledge expression model. To maximize the effectiveness of ontology, the relationships between classes, properties and instances are required. W3C provides principles about building, publishing and accessing ontology through LOD project and many ontologies such as FOAF, DBPedia, DBLP are cross-linked and being utilized actively.

In this study, we propose OntoURIResolver service for efficient instance integration of your own URIs and URIs in LOD by providing these functions; 1) User can get URIs by URI or entity name, 2) Service provides identified groups of URIs and recommends canonical URI and entity name for each group. 3) User can edit results generated by service and modified results by users will be used for URIs resolution next time. Currently, we collect URIs from sameas.org and sindice.com and gather RDF triples from SPARQL endpoints using web services. The resolved

types of our service are limited to author and article and we are expanding resolved types gradually.

2 Related Work

Most researches for entity identification have been carried out to disambiguate the authors of academic literatures by NLP (Natural language Processing) techniques [1,2,3]. Large evaluation dataset was constructed and information obtained through a web search was used to enhance the performance of author disambiguation [4]. But these researches have limitation on disambiguation type. And Scopus provides academic information service using author disambiguation [5].

Southampton Univ. develops and operates sameas.org service that provides co-referent information collected from LOD and other ontologies [6]. They gathered co-referent information by SPARQL endpoints and RDF dumps. It is very useful service, because user can search URI lists and co-referent information which are spared widely in the web through sameas.org. But, they do not create new URI group using RDF triples and the collected co-referent data contains incorrect information.

3 OntoURIResolver

OntoURIResolver is a URI resolving service that collects RDF triples in LOD, makes groups per entity and recommends canonical URI and entity name per each group in order to link our own ontology to LOD. OntoURIResolver uses OntoReasoner [7], which is a reasoning engine developed by KISTI, as triple store, Resolving process is executed in real-time because, RDF triples in LOD are frequently changed and the results depend on the status of network and external services. It collects RDF triples by using SPARQL endpoints and accessing URIs. And it uses triple store as a temporally to store collected RDF triples, mapping table of classes and properties between multiple ontologies, pre-executed resolved results and edited information by users.

Figure 1 shows URI resolution process. This process can be divided into six stages; 1) Collect RDFs of URIs: In this stage, URIs are classified into several groups according to status of URL, returned data types and status of URI. One is normal group containing URIs to be resolved and the other is abnormal group containing deprecated URIs, dead links and URIs that formed in non-RDF data. After this stage, only normal group will be treated at next steps. 2) Make groups by entity type: we can separate some URIs using information type of URIs. To improve the accuracy of type judgment, we constructed the relationships between classes. 3) Make groups by entity name: URIs in normal group can be divided into several subgroups based on similarity of entity names that is calculated with string match method. If the threshold is high then some URIs remains separately. And if the threshold is low then this method makes over-clustering. We have to enhance the accuracy of this method with various heuristics. 4) Collect additional properties of URIs: In this stage, OntoURIResolver collects additional RDF triples to divide URIs having same type

and same name. For resolving authors, we use some properties related to 'creator', 'author', 'articles created by someone', 'email', 'affiliation' and etc. To do this process, we constructs mapping table about relationships between properties of LOD. To expand resolving types, more information related to various types are required. RDF triples are obtained by SPARQL endpoints in real-time. 5) Compare multiples properties of URIs: RDF triples collected in previous stage is compared to each other. 6) Identify URIs: URIs are separated to some groups using previously calculated values.

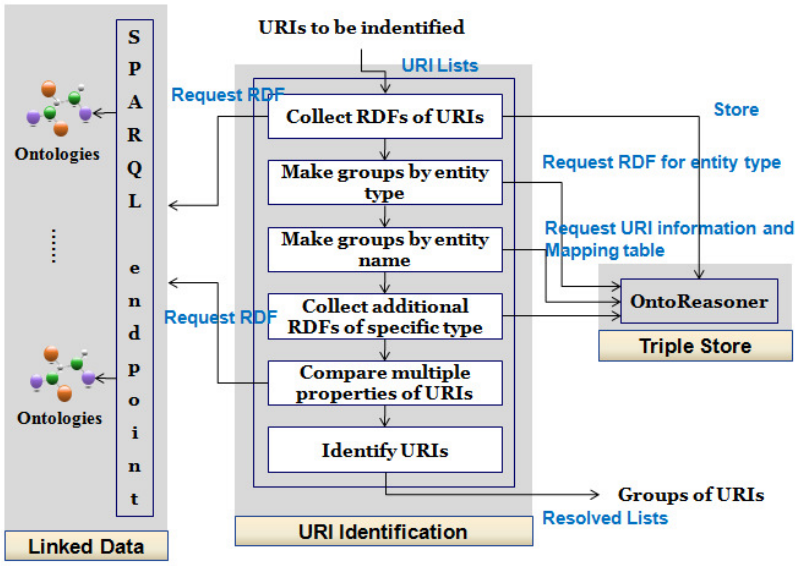

Figure1. Process of URI Resolution

Figure 2 is a snapshot of OntoURIResolver service. The service shows different results of OntoURIResolver and sameas.org. Actually, sameas.org does not try to merge URIs. They provide only co-referent information gathered by LOD. Of course, sameas.org provides very useful information about co-referent relationships between URIs. But OntoURIResolver classifies normal URIs and abnormal URIs and helps user to select URIs of specific entity. User can see RDF triples belonging to specific group by clicking button. To separate meaningful and useful URIs and others, URIs are divided into two groups. One is resolved URI list and the other is unresolved URI list. Unresolved URI list contains URIs which are inconsistency in label of URIs, deprecated URIs which are no longer used, non-rdf URIs which are URLs of microformat or html data and internal server error pages. By separating unresolved URI list, URIs to be resolved will be reduced.

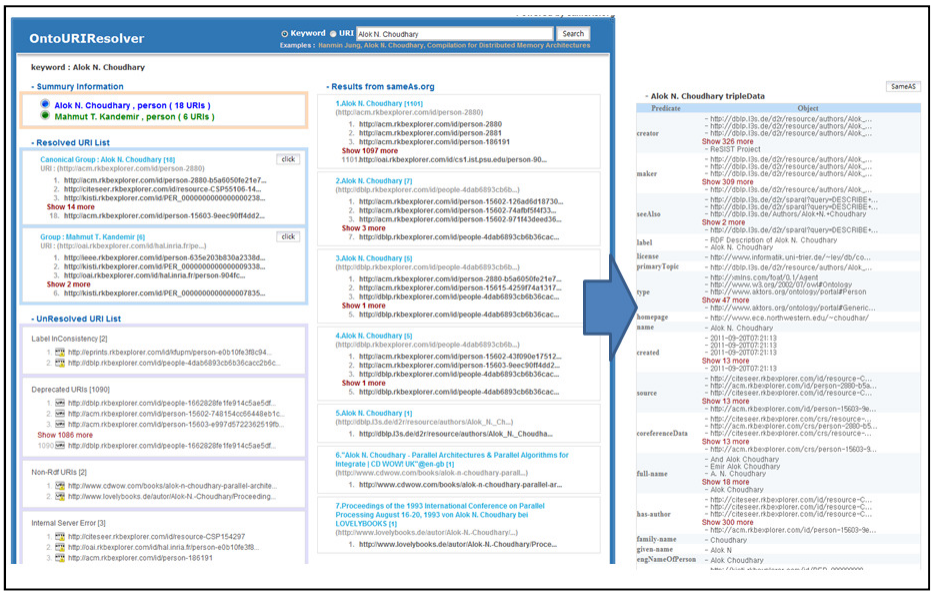

Figure2. User Interface of OntoURIResolver
(From http://our.kisti.re.kr)

4 Experiments and Evaluation

Our proposed method is experimented by using top 10 ranked authors of DBLP. Top ranked author means a person who has many publications in DBLP. Generally, there are many RDF triples and URIs related to top ranked authors. And then we compared our resolved results to sameas.org. The number of groups created by OntoURIResolver is smaller than sameas.org and the correctness of our results is checked manually.

Table 1. Comparison Table of sameas.org and OntoURIResolver

Author Name	sameas.org		OntoURIResolver		
	# of group	# of URI	# of group	# of valid URI	# of invalid URI
Philip S. Yu	4	690	1	11	689
Chin-Chen Chang	9	80	1	11	69
Elisa Bertino	8	532	1	15	517
Thomas S. Huang	10	28	1	5	23
Wen Gao	10	12	2	3	9
Wei Zhang	10	12	1	6	6
Edwin R. Hancock	3	570	2	9	561
Sudhakar M. Reddy	3	588	1	11	577
Ming Li	10	10	1	3	7
Li Zhang	10	10	2	6	4

5 Conclusions and Future Work

Ontology is very useful knowledge expression model for sharing, reusing and unification of multiple data sources. And LOD provides effective environment to interconnection among multiple ontologies. To make a local ontology interoperable and sharable with LOD, we have to make linkages URIs between local ontology and LOD. In order to decide appropriate URI for specific entity, OntoURIResolver collects RDF (Resource Description Framework) triples of multiple ontologies by SPARQL, divides URIs into several groups by comparing RDF triples of URIs and recommends a canonical URI and entity name for each group using statistics of RDF triples. We experiment comparison of sameas.org and OntoURIResolver with top 10 ranked authors of DBLP. Users can find a specific URI for entity and make interconnections with LOD to maximize the effectiveness of ontology through this service. In this study, we proposed OntoURIResolver service for efficient instance integration of your own URIs and URIs in LOD. The resolved types of our service are limited to author and article and we are expanding resolved types gradually.

References

1. A. L. Gentile, Z. Zhang, L. Xia, J. Iria: Semantic Relatedness Approach for Named Entity Disambiguation, J. Communications in Computer and Information Science, vol. 91-4, pp. 137-148. Springer, Heidelberg (2011)
2. A. Pilz, G. Paaß: Named entity resolution using automatically extracted semantic information, Entity 2010 Workshop at LREC 2010, Malta (2010)
3. E. Brill : Pattern-Based Disambiguation for Natural Language Processing, EMNLP/VLC-2000, HongKong (2000)
4. I. Kang, S. Na, S. Lee, H. Jung, P. Kim, W. Sung: On co-authorship for author disambiguation. J. Information Processing & Management, vol. 45-1, pp. 84-97. Elsevier, (2009)
5. Elsevier B.V., "SciVerse Scopus | SciVerse", http:// www.info.sciverse.com/scopus/, 2011
6. H. Glaser, A. Jaffri, T. Millard: Managing Co-reference on the Semantic Web, WWW2009 Workshop: Linked Data on the Web (LDOW2009), Madrid (2009)
7. S. Lee, M. Lee, P. Kim, H. Jung, W. Sung: OntoFrame S3: Semantic Web-Based Academic Research Information Portal Service Empowered by STAR-WIN. LNCS, vol. 6089, pp. 401-405. Springer, Heidelberg (2010)

Appendix

A. Minimal requirements

End-user application. OntoURIResolver is an application that is not only for experts in semantics domain but also for the general web users. It provides resolved URI groups and recommends canonical URI for specific object. And then our service is useful to interconnect local semantic data to linked open data.

Information sources. To identify URIs in liked open data, our system uses URI lists from sindice.com and sameAs.org, RDF triples acquired by SPARQL end point and heuristic rules about relationships between ontology properties. During disambiguation, every data is collected from web at real time. DBLP, citeseer, RKB explorer and ontologies related to publications are especially used at our service. These information sources are syntactically, structurally and semantically heterogeneous.

Meaning of data. The goal of OntoURIResolver is author disambiguation by comparing multiple ontology properties. There are too many equivalent URIs in LOD, and it is not easy to find appropriate URI to be interlinked with local URI. In order to find meaningful properties and values, we adapt FCA(Formal Concept Analysis) approach for analyzing ontologies and create an upper level ontology to integrate features from several different ontologies. We create heuristic rules about semantic relationships between multiple ontologies. OntoURIResolver stores information into a triple repository named OntoReasoner.

B. Additional features

User interface. OntoURIResolver provides simple and easy user interface. When user inputs URI or object name, service generates URI groups and recommends a canonical URI comparing to other service. In the near future, we will introduce new function which is for editing and managing disambiguated results at online.

Scalability. There is no limitation on data size. At now, the resolved type of OntoURIResolver is restricted to author and we will expand resolving types including institution, location and so on. The repository of our service is OntoReasoner and it is designed for large semantic data based on DBMS.

Commercial applicability. The disambiguated results will be provided in form of RDF triples and everyone can get RDF triples by SPARQL end point. There is no plan to commercialize OntoURIResolver.

Accuracy. We evaluate the accuracy of our system with top-10 authors in DBLP. In order to improve accuracy and speed, we use statistics and heuristic methods and these efforts will be continued.

Multilinguality. OntoURIResolver supports multiple languages such as English, Korean, Japanese and so on.

Range of devices. OntoURIResolver supports diverse devices including Web browser(including mobile browsers) such as PC, smart phone, PDA, etc except for MS Internet Explorer. Therefore, we planned to expand compatibility.